

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-229138

(43)Date of publication of application : 24.08.2001

(51)Int.Cl.

G06F 15/16

(21)Application number : 2000-041434

(71)Applicant : SONY CORP

(22)Date of filing : 15.02.2000

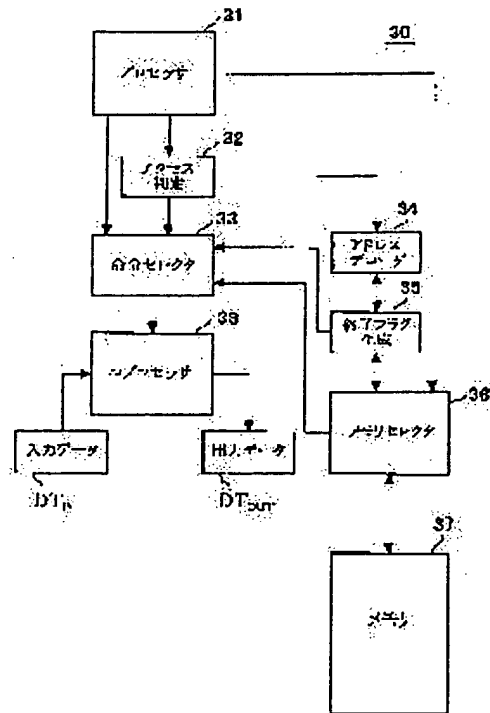
(72)Inventor : NOSE ATSUSHI

## (54) DEVICE AND METHOD FOR IMAGE PROCESSING

**(57)Abstract:**

**PROBLEM TO BE SOLVED:** To provide a device and a method for image processing which enable a processor to perform different operation while a coprocessor is issuing an instruction code and to shorten an image processing time.

**SOLUTION:** Instructions and data which are used by the processor 31 and coprocessor 38 and instructions and data which are read in directly by the coprocessor 31 are held in a memory 37. The processor 31 performs a specific processing for a given instruction which is an instruction to be executed by the coprocessor 38, and outputs the result, but it sets a direct read-in flag when the given instruction is an instruction to be read in directly by the coprocessor 38. Where the direct read-in flag is not set, the instruction outputted from the processor 31 is supplied to the coprocessor 38, but where the direct read-in flag is set, a directly read-in instruction in the memory 38 is supplied directly to the coprocessor 38.



## LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開2001-229138

(P2001-229138A)

(43)公開日 平成13年8月24日(2001.8.24)

(51)Int.Cl.<sup>7</sup>

G 0 6 F 15/16

識別記号

6 2 0

F I

G 0 6 F 15/16

テーム(参考)

6 2 0 G 5 B 0 4 5

審査請求 未請求 請求項の数8 OL (全 14 頁)

(21)出願番号 特願2000-41434(P2000-41434)

(22)出願日 平成12年2月15日(2000.2.15)

(71)出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72)発明者 能勢 敦

東京都品川区北品川6丁目7番35号 ソニー株式会社内

(74)代理人 100094053

弁理士 佐藤 隆久

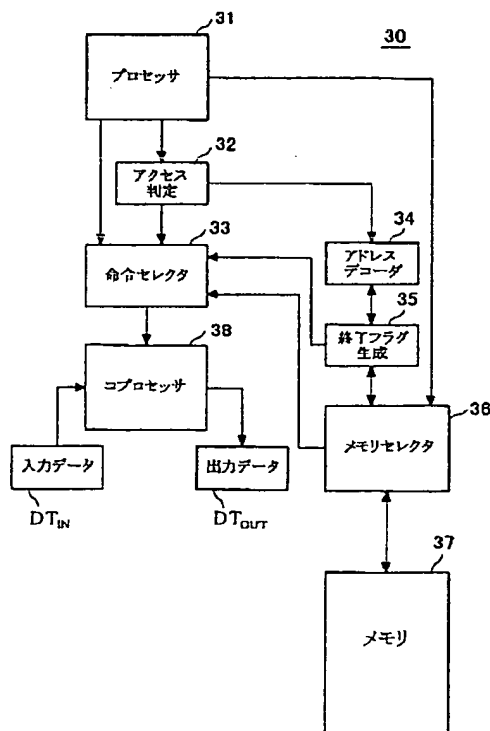
Fターム(参考) 5B045 AA01 BB48 GG09

(54)【発明の名称】 画像処理装置および画像処理方法

(57)【要約】

【課題】コプロセッサの命令コード発行中にプロセッサで別の演算が可能で、画像処理時間を短縮できる画像処理装置および画像処理方法を提供する。

【解決手段】プロセッサ31、およびコプロセッサ38で使用される命令、データ、並びにコプロセッサ31で直接読み込む命令、データをメモリ37に保持しておき、プロセッサ31において、与えられた命令が、コプロセッサ38で実行される命令である場合には当該命令に対する所定の処理を行って出力し、与えられた命令がコプロセッサ38で直接読み込む命令の場合には、直接読み込み用フラグを設定し、直接読み込み用フラグが設定されていない場合にはプロセッサ31から出力された命令をコプロセッサ38に供給し、直接読み込み用フラグが設定されている場合には、メモリ37の直接読み込み命令、データをコプロセッサ38に直接供給する。



**【特許請求の範囲】**

【請求項1】 入力された画像データに対して供給される命令に基づいて画像処理し、出力するコプロセッサと、

与えられた命令が、上記コプロセッサで実行される命令である場合には当該命令に対する所定の処理を行って出力し、与えられた命令が上記コプロセッサで直接読み込む命令の場合には、直接読み込み用フラグ設定要求を発行するプロセッサと、

上記直接読み込み用フラグ設定要求を受けると直接読み込み用フラグを設定するアクセス判定回路と、

上記プロセッサ、およびコプロセッサで使用される命令、データ、並びにコプロセッサで直接読み込む命令、データを保持するメモリと、

上記プロセッサによるアクセス要求を受けた場合には、メモリの命令、データを当該プロセッサに供給し、かつ上記アクセス判定回路により直接読み込み用フラグが設定されていない場合には上記プロセッサから出力された命令を上記コプロセッサに供給し、直接読み込み用フラグが設定されている場合には、上記メモリの上記直接読み込む命令、データを上記コプロセッサに供給する選択手段とを有する画像処理装置。

【請求項2】 上記アクセス判定回路により直接読み込み用フラグが設定されている場合には、上記メモリのアクセス要求を上記選択手段に対して発行するアクセス要求手段とを有し、

上記選択手段は、上記アクセス要求手段によるアクセス要求が発行されている場合に、上記メモリの上記直接読み込む命令、データを上記コプロセッサに供給する請求項1記載の画像処理装置。

【請求項3】 上記アクセス要求手段は、上記選択手段にメモリから読み出された上記直接読み込む命令を受けてデコードし、デコードの結果終了命令である場合には終了フラグを設定し、

上記アクセス判定回路は、終了フラグが設定された場合には、上記直接読み込み用フラグの設定を解除する請求項2記載の画像処理装置。

【請求項4】 上記選択手段は、上記プロセッサによるアクセス要求と上記アクセス要求手段によるアクセス要求を同時に受けた場合には、上記プロセッサによるアクセス要求を優先する請求項2記載の画像処理装置。

【請求項5】 上記選択手段は、上記プロセッサによるアクセス要求と上記アクセス要求手段によるアクセス要求を同時に受けた場合には、上記プロセッサによるアクセス要求を優先する請求項3記載の画像処理装置。

【請求項6】 上記コプロセッサは、1次元的に多並列に配置され、制御コードに従って供給される画像データに対する所定の演算処理を行う複数の要素プロセッサと、アプリケーションプログラムに基づいて上記複数の要素プログラムを連動して制御する制御コードを生成す

るプログラム制御部とを有する請求項1記載の画像処理装置。

【請求項7】 上記各要素プロセッサは、画像データを入力するための手段を複数備えた入力シリアルアクセスメモリ部と、入力データ、演算結果を保持するデータメモリ部と、演算を行うALUアレイ部と、出力画像データを取得するための手段を複数備えた出力シリアルアクセスメモリ部との一部を含む請求項6記載の画像処理装置。

【請求項8】 入力された画像データに対して供給される命令に基づいて画像処理し、出力するコプロセッサと、上記コプロセッサで実行される命令を供給、制御するプロセッサを用いた画像処理方法であって、

上記プロセッサ、およびコプロセッサで使用される命令、データ、並びにコプロセッサで直接読み込む命令、データをメモリに保持しておき、

上記プロセッサにおいて、与えられた命令が、上記コプロセッサで実行される命令である場合には当該命令に対する所定の処理を行って出力し、与えられた命令が上記コプロセッサで直接読み込む命令の場合には、直接読み込み用フラグを設定し、

直接読み込み用フラグが設定されていない場合には上記プロセッサから出力された命令を上記コプロセッサに供給し、直接読み込み用フラグが設定されている場合には、上記メモリの上記直接読み込む命令、データを上記コプロセッサに直接供給する画像処理方法。

**【発明の詳細な説明】****【0001】**

【発明の属する技術分野】本発明は、たとえばデジタル処理システム等に適用される画像処理装置および画像処理方法に関するものである。

**【0002】**

【従来の技術】画像データは、音声データ等の他のデータに比べ情報量が多く、一度に多くの処理を必要とする。そして、画像信号処理では1枚の画像を構成する全ての画素に対して同様の演算処理を施すことが多い。多くのデータに対して同様の演算処理を高速に実行するために、SIMD (Single Instruction Multiple Data Stream、単一命令複数データ) 型アーキテクチャが提案され、画像信号処理に限らず広い分野で利用されている。

【0003】SIMD型アーキテクチャは、演算装置に必要な個数だけ並べて、各々の演算装置が同一の命令に従って動作するようにした構成である。したがって、各々の演算装置に別々のデータを与えると、それぞれのデータに対する演算結果が一度に得られる。

【0004】SIMD型処理装置の画像処理への適用としては、たとえば「Kurokawa et al., "5.4GOPS Linear Array Architecture DSP for Video-Format Conversion", IEEE 1996/Feb. ISSCC, FP 15.7.」に示される装置

が知られている。

【0005】図7は、SIMD型画像処理プロセッサの構成例を示すブロック図である。このSIMD型画像処理プロセッサ10は、入力ポインタ11、入力SAM（シリアルアクセスメモリ）部12、データメモリ部13、ALUアレイ部14、出力SAM部15、出力ポインタ16、およびプログラム制御部17により構成される。

【0006】SIMD型画像処理プロセッサ10においては、入力SAM部12、データメモリ部13、ALUアレイ部14、および出力SAM部15により、全体でリニアアレイ（直線配列）型に多数並列化された要素プロセッサ群が構成されている。この多数の要素プロセッサPEは、プログラム制御部17内にある共通の一つの命令コードにより連動して制御（SIMD制御）される。プログラム制御部17には、プログラムメモリとそのプログラムの歩進のためのシーケンス制御回路などがあり、プログラムメモリにあらかじめ書き込まれたプログラムに従って接続されている各部分に対する各種制御信号を発生して、各部分を制御する。

【0007】図7に示すSIMD型画像処理プロセッサ10による画像処理は次のような手順で行われる。水平走査アクティブ期間内に、入力SAM部12に1水平走査線分の入力データを取り込み、水平走査部ブランキング期間で入力SAM部12からデータメモリ部13への転送が行われる。データメモリ部13、ALUアレイ部14では、プログラムに応じて演算処理が行われる。ALUアレイ部14における演算処理終了後、処理結果が出力SAM部15に転送され、水平走査アクティブ期間内に、出力SAM部15から1水平走査線分データが出力される。上記の処理中に、各部はすべて並列に動作している。

【0008】このような要素プロセッサ群をアレイ状に配列した並列プロセッサで画像処理を行うためには、特にSIMD制御の場合、左右の要素プロセッサとの通信機能が不可欠となる。この左右通信（今後は、LR通信、またはLRと記述する）機能により、フィルタ演算や、補間演算といった左右の画素の処理を可能としている。

【0009】ところで、画像処理をするSIMD型の処理装置は複雑な処理を必要とするために、多数の命令コードの発行を必要とし、その制御は年々複雑となっている。そこで、プログラム制御部に命令コードを効率良く自動的に発行するために、プロセッサが使われるようになった。このプロセッサは汎用CPU等の一般的なプロセッサであって、SIMD型画像処理装置はこのプロセッサに対して、コプロセッサと呼ばれる。

【0010】そして、コプロセッサにおけるプログラム制御部によって発生される制御信号は、プロセッサより得られる命令コードをデコードすることで実現する。制

御するための命令コードのパラメータ値や演算の種類等の情報をプロセッサで処理し、コプロセッサに供給する。たとえば、命令コードで同一の演算命令が続くとき、パラメータを変えるだけで、係数を変更できるとで、データメモリの使用領域を削減できる利点がある。

【0011】コプロセッサで使用される命令コードに対してプロセッサで処理を施すためには、メモリ（データ記憶装置）のデータ領域に格納される必要がある。プロセッサのデータメモリにコプロセッサの命令コードも格納されるため、プロセッサとコプロセッサはメモリを共有する必要がある。この構成を単一メモリ構成と呼ぶ。プロセッサとコプロセッサには従属関係があり、コプロセッサに命令コード（プロセッサにとってはデータ）を供給するためにはプロセッサがデータを取得し、コプロセッサに供給する必要がある。

【0012】また、コプロセッサの命令コードの中にはプロセッサで処理する必要はなく、そのままコプロセッサに供給されるものも存在する。プロセッサにおける処理を必要としない命令はプロセッサに一度読み込まれて、そのままコプロセッサへ供給されることになる。この場合、処理を必要としないコプロセッサ命令の発行が続く場合、プロセッサはメモリからデータを読んで、コプロセッサにデータを供給するためだけに使われることになる。プロセッサは、時間かかる係数計算があった場合でも、コプロセッサへの命令供給が終わるまで待つ必要がある。

【0013】図8は、プロセッサ、およびコプロセッサを用いた従来の画像処理装置のシステム構成例を示すブロック図である。この画像処理装置20は、図8に示すように、プロセッサ21、コプロセッサ22、およびメモリ23を有する。

【0014】図8において、SIMD型画像処理装置であるコプロセッサ22への入力データDTINは、実際に画像処理を行う基となる画像データである。また、出力画像データDTOUTは、処理後の画像データである。また、データのフォーマットは、コンピュータディスプレイで代表される色の3原色R（赤）、G（緑）、B（青）に基づくRGB形式、あるいはNTSCで代表されるテレビ信号があるが、特定のフォーマットに限定されない。

【0015】プロセッサ21は、メモリ23に対して、データの読みだし要求（リード要求）、データが格納してあるアドレスを供給する。データを保持する場合には、データの書き込み要求（ライト要求）、データを格納するアドレスおよびデータを供給する。メモリ23はリード要求を受けて、指定されるアドレスに格納されているデータをプロセッサに戻す。このときにデータを出したことを表す信号であるイネーブル信号をプロセッサ21に返す。ライト要求の場合、指定されたアドレス番地にデータを格納する。プロセッサ21はメモリ23

から受け取ったデータを必要な場合は演算し、コプロセッサ 22 の命令コードを生成する。生成された命令コードはコプロセッサ 22 に供給される。コプロセッサ 22 は、供給された命令コードに従い、入力データ DTIN に対する画像処理を行う。また、コプロセッサ 22 は、プロセッサ 21 の持つ命令を繰り返し実行し、全ての命令の実行が終了した時点で出力データ DTOUT を出力する。

【0016】次に、図 8 の画像処理装置におけるメインプロセスについて、図 9 のフローチャートに関連付けて説明する。

【0017】まず、ステップ ST1 において、入力画像データ DTIN をコプロセッサ 22 に読み込む。次に、ステップ ST2 において、プロセッサ 21 で実行する命令をリードする。プロセッサ 21 は、ステップ ST3 において、リードしてきた命令が、コプロセッサ 22 へ命令コードを発行するための命令（単一命令コード発行命令）でない場合にはステップ ST7 の処理に移行する。なお、単一命令コード発行とは、メモリ 23 に格納されているデータの中の 1 命令コードをプロセッサ 21 を通してコプロセッサ 22 に供給することをいう。

【0018】ステップ ST3 において、コプロセッサ 22 への単一命令コードの発行命令であると判断した場合には、プロセッサ 21 は、ステップ ST4 において、メモリ 23 からデータを取得する。このデータはそのままコプロセッサ 22 の命令コードとなる。次に、ステップ ST5 において、プロセッサ 21 は、取得したデータをコプロセッサ 22 に供給する。そして、ステップ ST6 において、プロセッサ 21 の命令のカウントであるプログラムカウンタをインクリメントし、ステップ ST2 の処理に戻る。

【0019】また、ステップ ST7 において、リードした命令がプロセッサ 21 の終了命令である場合、ステップ ST18 の処理に移行する。一方、ステップ ST7 でリードした命令がプロセッサ 21 の終了命令でないと判断した場合には、プロセッサ 21 は、ステップ ST8 においてリードした命令を実行する。そして、ステップ ST9 において、プロセッサ 21 の命令のカウントであるプログラムカウンタをインクリメントし、ステップ ST2 の処理に戻る。また、ステップ ST10 において、コプロセッサ 22 の動作が終了しているのを確認して、画像データを出力する。

【0020】図 10 は、従来の画像処理装置 10 の命令の実行順序の一例を示す図である。図 10 に示すように、従来の画像処理装置 10 においては、命令 1～命令 7、およびコプロ命令 1～コプロ命令 5 が、所定の順序で順番に行われる。

【0021】

【発明が解決しようとする課題】 上述したように、コプロセッサ 22 の命令コードの発行はプロセッサ 21 によ

って行われる。命令コードはプロセッサ 21 で演算した結果を利用して生成できる。たとえば、コプロセッサ 21 で行われる演算の係数をプロセッサで計算し、定数値としてコプロセッサ 22 に供給することができる。

【0022】ところで、命令コードの全てがプロセッサ 21 で演算してからコプロセッサ 22 に供給する必要はない。メモリ 23 に格納されているデータをそのままコプロセッサ 22 に供給する場合もある。そのような命令コードが多く存在するプログラムにおいて、プロセッサ 21 はコプロセッサ 22 の命令発行のために実行を繰り返すことから、他の演算処理を実行できないという問題が生じる。たとえば、時間のかかる演算をしてコプロセッサ 22 の係数を算出する必要があったときに、コプロセッサ 22 に命令発行をしている間に演算ができないために、画像処理全体の処理速度の低下の問題が生じる。

【0023】本発明は、かかる事情に鑑みてなされたものであり、その目的は、コプロセッサの命令コード発行中にプロセッサで別の演算が可能で、画像処理時間を短縮できる画像処理装置および画像処理方法を提供することにある。

【0024】

【課題を解決するための手段】 上記目的を達成するため、本発明は、入力された画像データに対して供給される命令に基づいて画像処理し、出力するコプロセッサと、与えられた命令が、上記コプロセッサで実行される命令である場合には当該命令に対する所定の処理を行って出力し、与えられた命令が上記コプロセッサで直接読み込む命令の場合には、直接読み込み用フラグ設定要求を発行するプロセッサと、上記直接読み込み用フラグ設定要求を受けると直接読み込み用フラグを設定するアクセス判定回路と、上記プロセッサ、およびコプロセッサで使用される命令、データ、並びにコプロセッサで直接読み込む命令、データを保持するメモリと、上記プロセッサによるアクセス要求を受けた場合には、メモリの命令、データを当該プロセッサに供給し、かつ上記アクセス判定回路により直接読み込み用フラグが設定されていない場合には上記プロセッサから出力された命令を上記コプロセッサに供給し、直接読み込み用フラグが設定されている場合には、上記メモリの上記直接読み込む命令、データを上記コプロセッサに供給する選択手段とを有する。

【0025】また、本発明では、上記アクセス判定回路により直接読み込み用フラグが設定されている場合には、上記メモリのアクセス要求を上記選択手段に対して発行するアクセス要求手段とを有し、上記選択手段は、上記アクセス要求手段によるアクセス要求が発行されている場合に、上記メモリの上記直接読み込む命令、データを上記コプロセッサに供給する。

【0026】また、本発明では、上記アクセス要求手段は、上記選択手段にメモリから読み出された上記直接読

み込む命令を受けてデコードし、デコードの結果終了命令である場合には終了フラグを設定し、上記アクセス判定回路は、終了フラグが設定された場合には、上記直接読み込み用フラグの設定を解除する。

【0027】また、本発明では、上記選択手段は、上記プロセッサによるアクセス要求と上記アクセス要求手段によるアクセス要求を同時に受けた場合には、上記プロセッサによるアクセス要求を優先する。

【0028】また、本発明では、上記コプロセッサは、1次元的に多並列に配置され、制御コードに従って供給される画像データに対する所定の演算処理を行う複数の要素プロセッサと、アプリケーションプログラムに基づいて上記複数の要素プログラムを連動して制御する制御コードを生成するプログラム制御部とを有する。

【0029】また、本発明では、上記各要素プロセッサは、画像データを入力するための手段を複数備えた入力シリアルアクセスメモリ部と、入力データ、演算結果を保持するデータメモリ部と、演算を行うALUアレイ部と、出力画像データを取得するための手段を複数備えた出力シリアルアクセスメモリ部との一部を含む。

【0030】また、本発明は、入力された画像データに対して供給される命令に基づいて画像処理し、出力するコプロセッサと、上記コプロセッサで実行される命令を供給、制御するプロセッサを用いた画像処理方法であって、上記プロセッサ、およびコプロセッサで使用される命令、データ、並びにコプロセッサで直接読み込む命令、データをメモリに保持しておき、上記プロセッサにおいて、与えられた命令が、上記コプロセッサで実行される命令である場合には当該命令に対する所定の処理を行って出力し、与えられた命令が上記コプロセッサで直接読み込む命令の場合には、直接読み込み用フラグを設定し、直接読み込み用フラグが設定されていない場合には上記プロセッサから出力された命令を上記コプロセッサに供給し、直接読み込み用フラグが設定されている場合には、上記メモリの上記直接読み込む命令、データを上記コプロセッサに直接供給する。

【0031】本発明によれば、プロセッサにおいて、与えられた命令が、コプロセッサで実行される命令である場合には当該命令に対する所定の処理が行われて選択手段に出力される。一方、与えられた命令がコプロセッサで直接読み込む命令の場合には、直接読み込み用フラグがアクセス判定回路で設定される。そして、直接読み込み用フラグが設定されていない場合にはプロセッサから出力された命令がコプロセッサに供給され、直接読み込み用フラグが設定されている場合には、メモリの上記直接読み込む命令、データをコプロセッサに直接供給される。したがって、コプロセッサの命令発行中にプロセッサで別の演算が可能となるため、同一処理時間内にプロセッサで多くの演算を実現できる。プロセッサを通さずに命令を発行できるため、コプロセッサに早く命令を発

行できる。コプロセッサに対して命令の発行数が増えるため、多くの画像処理演算を実現できる。その結果、画像処理時間を短縮できる。

【0032】

【発明の実施の形態】図1は、本発明に係る画像処理装置の一実施形態を示すブロック図である。

【0033】本画像処理装置30は、プロセッサ31、アクセス判定回路32、命令セクタ33、アドレスデコーダ34、終了フラグ生成回路35、メモリセクタ36、メモリ37、およびコプロセッサ38により構成されている。なお、アドレスデコーダ34、および終了フラグ生成回路35によりアクセス要求手段が構成され、命令セクタ33、およびメモリセクタ36により選択手段が構成される。

【0034】プロセッサ31は、基本的には、コプロセッサ38で実現される処理を制御する。具体的には、プロセッサ31は、画像データを処理するためのコプロセッサ38の命令コードを発行する。また、プロセッサ31は、メモリセクタ36にリード要求、データおよびアドレスを出す。そして、プロセッサ31は、リード時は、メモリ37に格納されているデータをメモリセクタ36より受け取る。受け取ったデータがコプロセッサ38の命令コードである場合、プロセッサ31は、命令コードの係数演算や割り込み処理等の判定をし、命令セクタ33に命令コードを供給する。

【0035】たとえば、プロセッサ31の持つ変数をx、コプロセッサ38のデータメモリ部をMemとすると、たとえば、変数xとコプロセッサ38のデータメモリ部の4番地に入っている内容を加算する命令は、次のようになる。

$Mem[3] = Addi(Mem[4], x)$

【0036】メモリ27では、次の情報を持つ。

$COP(Mem[3] = Addi(Mem[4], x))$

ここで、COPは、コプロセッサ38に発行する命令であることを示す。

【0037】プロセッサ31は、コプロセッサ38に発行する命令であることを、このCOPで判断し、変数xの値を代入して命令セクタ33に命令コードとして供給する。たとえば変数xが3の場合、命令は次のような形態で、命令セクタ33に供給される。

$Mem[3] = Addi(Mem[4], 3)$

【0038】このとき同時に、プロセッサ31は、プロセッサイネーブル信号を命令セクタ33に出力する。また、コプロセッサ38の命令コードを直接メモリ37からコプロセッサ38に読み込む命令がプロセッサ31に与えられたとき、プロセッサ31は、アクセス判定回路32に対して、直接リード用のアクセスフラグ(直接読み込み用フラグ)を設定する(アクティブにする)要求を出す。

【0039】メモリ37から直接コプロセッサ38に読み込む命令は、たとえば次の形成形で表される。

COPREAD (Address)

ここで、Addressは、コプロセッサ38に発行する命令コードの格納されているメモリ37の始めの番地(スタートアドレス)である。

【0040】プロセッサ31は、COPREADを識別子として、アクセスフラグを設定する要求とスタートアドレスをアクセス判定回路32に出す。アクセスフラグが立っている間は、プロセッサ31は、コプロセッサ38への命令コード発行以外の命令、すなわちプロセッサ31内で処理できる命令を実行できるものとする。また、プロセッサ31は、アクセスフラグが立っている時は命令セクタ33に命令を供給できない。アクセスフラグが立っている時に命令セクタ33に命令コードを供給する、もしくはアクセス判定回路32にアクセス要求を出す場合、プロセッサ31はアクセスフラグが下がるまで、停止する。

【0041】アクセス判定回路32は、プロセッサ31より直接リード用の要求を受けた場合、アクセスフラグを立てる。そして、アクセス判定回路32は、アクセスフラグが立っていることを命令セクタ33、アドレスデコーダ34に通達し、アドレスデコーダ34にスタートアドレスを供給する。また、アクセス判定回路32は、アドレスデコーダ34より終了フラグを受け取った場合、アクセスフラグを下げる。なお、アクセスフラグは、メモリ37からコプロセッサ38へ連続命令コード発行が実行されていることを表すフラグである。

【0042】命令セクタ33は、アクセス判定回路32によるアクセスフラグ(直接読み込み用フラグ)が立っている場合はメモリセクタ36よりデータを受け取り、アクセスフラグが下がっている場合はプロセッサ31からのデータを受ける。また、命令セクタ33は、アクセスフラグは立っていない場合にはプロセッサ31の、アクセスフラグが立っている場合にはメモリセクタ36のイネーブル信号が立っている場合のみ、コプロセッサ38に命令を供給する。

【0043】アドレスデコーダ34は、メモリ37に格納されているデータのアドレスを生成するセクタである。アドレスデコーダ34は、プロセッサ31からアクセス判定回路32に新たに命令コード連続発行の要求が出た場合は、アクセス判定回路32より受け取った新規のアドレスを終了フラグ生成回路35に供給する。アドレスデコーダ34は、すでにアクセスフラグが立っており、終了フラグが立っていない場合は、アドレスをインクリメントし、次のデータに対する要求を出す。インクリメントされたアドレスは終了フラグ生成回路35に供給される。同様に、アドレスデコーダ34は、終了フラグ生成回路35にアクセスフラグを渡す。

【0044】終了フラグ生成回路35は、アドレスデコ

ーダ34より得られたアクセスフラグおよびアドレスをメモリセクタ36に供給する。終了フラグ生成回路35は、メモリセクタ36より得られた命令コードの一部をデコードし、終了命令の場合には終了フラグを立てる。終了フラグはアドレスデコーダ34およびメモリセクタ36に供給される。終了フラグ生成回路35は、たとえば、次のような命令コードがメモリ37から得られた場合、終了フラグを立てる。

COPEND ()

【0045】メモリセクタ36は、プロセッサ31からのリード、ライト要求、データおよびアドレスと、終了フラグ生成回路35からのアクセス要求およびアドレスを受け取り、メモリ37にリード、ライト要求、データおよびアドレスを渡す。メモリセクタ36は、メモリ37より得られたデータはプロセッサ31からのアクセス要求と終了フラグ生成回路35より得られたアクセス要求を判定し、アクセス要求を受けた先にデータとイネーブル信号を返す。メモリセクタ36は、プロセッサ31と終了フラグ生成回路35の両方から要求等を同時に受けた場合には、プロセッサ31の要求等を優先する。メモリセクタ36は、メモリ37より得られた命令が「COPEND」の場合は、終了フラグ生成回路35に渡し、ここでデコードされた時点で、終了フラグを受け取る。このとき、メモリセクタ36は、コプロセッサ38で実現される命令ではないため、イネーブル信号は立てない(アクティブにしない)。

【0046】メモリ37は、プロセッサ31、コプロセッサ38で使用される命令、データを格納する。メモリ37は、メモリセクタ36よりリードの要求とアドレスを受けた場合には、指定されたアドレスに格納されている値をメモリセクタ36に返す。また、メモリ37は、ライトの要求とアドレスとデータを受けた場合には、指定されたアドレスにデータを書き込む。なお、コプロセッサ命令のリード要求、プロセッサデータのリード要求の区別はメモリ37では行わず、メモリセクタ36の段階で分けられる。また、プロセッサ31を通してコプロセッサ38に命令コードを発行する場合には、次のようになる。

COP (コプロセッサ命令コード)

これに対して、メモリセクタ36より直接コプロセッサ38に供給する場合には、COPを削減する必要がないため、次の形態で格納される。

コプロセッサ命令コード

【0047】また、メモリ37においては、コプロセッサ38用の連続発行する命令コードは、たとえば図2に示すような順序で格納されている。なお、図2において、ADD、SUB、MULはコプロセッサ38で実現される命令コードとする。

【0048】コプロセッサ38は、命令セクタ33により供給される命令コードに従い、入力データDTINIに

対する画像処理を行う。そして、コプロセッサ38は、命令セクタ31から供給される命令を繰り返し実行し、全ての命令の実行が終了した時点で出力データDTOUTを出力する。

【0049】コプロセッサ38は、たとえば図3に示すようなSIMD型画像処理プロセッサにより構成される。

【0050】このSIMD型画像処理プロセッサ380は、図3に示すように、入力ポインタ381、第1入力SAM（シリアルアクセスメモリ）部382a、第2入力SAM部382b、データメモリ部383、ALUアレイ部384、第1出力SAM部385a、第2出力SAM部385b、出力ポインタ386、およびプログラム制御部387を有している。また、本実施形態では、1画素に対応するそれぞれのブロックを合わせたものを要素プロセッサPEと呼ぶ。本SIMD型画像処理プロセッサ380は、要素プロセッサPEをアレイ状に画像のサイズ分配置した要素プロセッサ群より構成されている。また、本実施形態では、入力SAM部、出力SAM部を2つ有する構成を示しているが、本発明を実現するためには、これら入力SAM部、出力SAM部は、1つであっても、2つ以上の複数個持っていても良い。なお、SIMD型画像処理プロセッサ380は、ハードウェアで構成されていても良いし、ソフトウェアで実現されていても良い。

【0051】入力ポインタ381は、ALUアレイ部384の書き込みアドレスが、入力ポインタ381を指しているときに、ALUアレイ部384の演算結果を受けて入力SAM部382a、入力SAM部382bを制御する。なお、入力SAM部382aと入力SAM部382b制御用の入力ポインタを別々に持っていても良いし、同じポインタアドレスであっても良い。

【0052】入力SAM部382aは、入力ポインタ381の指示に従ってスキップ等の処理を含めて、第1入力フレームメモリ21に格納された画像データを所定の位置に入力し、そのデータをデータメモリ部383に供給する。入力SAM部382bは、入力ポインタ381の指示に従ってスキップ等の処理を含めて、第2入力フレームメモリ22に格納された画像データを所定の位置に入力し、そのデータをデータメモリ部383に供給する。

【0053】データメモリ部383は、1要素プロセッサPE当たり数百ビットのメモリから構成されており、プログラム制御部387の制御コードに従い、入力SAM部382a、382bからのデータを保持したり、ALUアレイ部384が演算を行うためのデータをALUアレイ部384に送り、また、ALUアレイ部384の演算結果を保持したりする。

【0054】ALUアレイ部384は、データメモリ部383からのデータを入力し、プログラム制御部387

で指定されたアドレスポインタに演算結果を書き込む。アドレスポインタがデータメモリ部383であった場合は、データメモリ部383の指定されたアドレスにALUアレイ部384の演算結果が書き込まれる。また、アドレスポインタが入力ポインタ381、出力ポインタ386であった場合には、入力ポインタ381、出力ポインタ382にALUアレイ部384の演算結果が書き込まれる。

【0055】出力SAM部385aは、プログラム制御部387の制御コードの書き込みアドレスが出力SAM部385aを指している場合、ALUアレイ部384の演算結果を入力し、第1出力フレームメモリ23へデータを供給する。出力SAM部385bは、プログラム制御部387の制御コードの書き込みアドレスが出力SAM部385bを指している場合、ALUアレイ部384の演算結果を入力し、第2出力フレームメモリ24へデータを供給する。なお、本実施形態では、出力SAM部を2つ持っているが、任意の画素数変換を説明するためには、出力SAM部は、1つであっても良いし、2つ以上の複数持っていても良い。

【0056】出力ポインタ386は、ALUアレイ部384の書き込みアドレスS384bが出力ポインタ386を指しているときに、ALUアレイ部384の演算結果を入力とする出力SAM部385a、および出力SAM部385bを制御する。なお、出力SAM部385aと出力SAM部385b制御用の出力ポインタを別々に持っていても良いし、同じポインタアドレスであっても良い。

【0057】プログラム制御部387は、所定の制御コードに従い、入力SAM部382a、382b、データメモリ部383、ALUアレイ部384、出力SAM部385a、385bを制御する制御コード出力S387を生成する。

【0058】要素プロセッサPEは、入力SAM部382a、382b、データメモリ部383、ALUアレイ部384、出力SAM部385a、385bの1画素に対応したものの集合のことであり、プロセッサは、この要素プロセッサが複数集まって構成されている。すなわち、普通のプロセッサでは、そのハードウェアは一般にワード処理プロセッサであり、ワードを単位として処理するが、図3の斜線で示した縦の細長い範囲で示すひとつの要素プロセッサPEは、入力SAM部12、データメモリ部13、出力SAM部15はメモリの「カラム」になっており、またALUアレイ部14は1ビットALUであり、事実上フルアダー（全加算器）を主体にした回路になっている。そのため、普通のプロセッサとは違ってビット処理プロセッサであり、ビットを単位として処理する。普通のCPUで言う8ビットマシンとか16ビットマシンという言い方に対応させれば1ビットマシンである。ビット処理プロセッサはハードウェアが小さ



く、普通には実現できない程多数の並列数を実現できるので、画像用の場合、要素プロセッサ直線配列の並列数は、映像信号の一水平走査期間の画素数(H)に一致させている。

【0059】図3に示すSIMD型画像処理プロセッサ380による画像処理は次のような手順で行われる。水平走査アクティブ期間内に、入力SAM部382a、382bに1水平走査線分の入力データDTIN1、DTIN2を取り込み、水平走査部ブランキング期間で入力SAM部382a、382bからデータメモリ部383への転送が行われる。データメモリ部383、ALUアレイ部384では、プログラムに応じて演算処理が行われる。ALUアレイ部384における演算処理終了後、処理結果が出力SAM部385a、385bに転送され、水平走査アクティブ期間内に、出力SAM部385a、385bから1水平走査線分データが出力される。上記の処理中に、各部はすべて並列に動作している。

【0060】このような要素プロセッサ群をアレイ状に配列した並列プロセッサで画像処理を行うためには、特にSIMD制御の場合、左右の要素プロセッサとの通信機能が不可欠となる。この左右通信(今後は、LR通信、またはLRと記述する)機能により、フィルタ演算や、補間演算といった左右の画素の演算処理を可能としている。また、要素プロセッサ間の通信のことを左右通信(LR通信)と呼び、左右隣あった画素の演算等を可能にする。

【0061】次に、上記構成を有する画像処理装置30の動作を、図4および図5のフローチャートに関連付けて説明する。図4は全体の動作の流れであるメインプロセスを示し、図5はメインプロセスが実行中に並列して動作するメモリからコプロセッサへのデータの供給の動作を表すサブプロセスを示している。

【0062】まず、ステップST101において、入力画像データDTINをコプロセッサ31に読み込む。次に、ステップST102において、プロセッサ31で実行する命令をリードする。プロセッサ31は、ステップST103において、リードしてきた命令が、コプロセッサ38へ命令コードを発行するための命令(単一命令コード発行命令)でない場合にはステップST107の処理に移行する。なお、単一命令コード発行とは、メモリ37に格納されているデータの中の1命令コードをプロセッサ31を通してコプロセッサ38に供給することをいう。一方、連続命令発行とは、プロセッサ31を介さずにメモリ37から直接プロセッサにデータを供給することをいう。

【0063】ステップST103において、コプロセッサ382への単一命令コードの発行命令であると判断した場合には、プロセッサ31は、ステップST104において、メモリセクタ36にリード要求、データおよびアドレスを出し、メモリ37に格納されているデータ

をメモリセクタ36より受け取る。受け取ったデータがコプロセッサ38の命令コードである場合、プロセッサ31は、命令コードの係数演算や割り込み処理等の判定をし、ステップST105において、命令コードを命令セクタ33に供給する。このとき、プロセッサ31は、プロセッササイネーブル信号を命令セクタ33に出力しており、また、アクセス判定回路32によるアクセスフラグが下がっている。したがって、命令セクタ33はプロセッサ31からのデータを受けて、コプロセッサ38に供給する。そして、ステップST106において、プロセッサ31の命令のカウンタであるプログラムカウンタをインクリメントし、ステップST102の処理に戻る。

【0064】また、ステップST107において、コプロセッサ38への連続発行命令であるか否かの判別を行う。そして、コプロセッサ38への連続発行命令でないと判断した場合、ステップST115の処理に移行し、コプロセッサ38への連続発行命令であると判断した場合、ステップST108においてサブプロセスを実行する。このときに、プロセッサ31は、アクセスフラグを立てる要求である実行フラグを立て、かつスタートアドレスをアクセス判定回路32に供給する。これにより、アクセス判定回路32は、アクセスフラグを立て、アクセスフラグが立っていることを命令セクタ33、アドレスデコーダ34に通達し、アドレスデコーダ34にスタートアドレスを供給する。

【0065】サブプロセスは、メインプロセスにて起動命令が発行されたときに動作を開始する。図5に示すように、ステップST1081において、コプロセッサ38が、メモリ37から、メモリセクタ36、命令セクタ33を介してデータをリードする。そして、ステップST1082でリードしたデータをデコードし、ステップST1083でアドレスをデコードする。ここで、ステップST1084において、終了命令であった場合には終了コードを発行する。終了命令でない場合はステップST1081の処理に戻る。

【0066】そして、ステップST109において、ストップ信号が発行され、ステップST110において終了フラグ生成回路35が終了フラグを生成すると、ステップST102の処理に戻る。

【0067】一方、ステップST109において、ストップ信号が発行されていない場合には、ステップST111の処理に移行する。ステップST111において、リードしたプロセッサ31にて実行すべき命令を実行する。このとき、プロセッサ31で実行される制御命令がコプロセッサ38への発行命令である場合(単一命令コード発行命令、および連続命令コード発行)、ストップ信号を発行する。そして、ステップST112において、プロセッサ31の命令のカウンタであるプログラムカウンタをインクリメントする。次いで、ステップST

113において、サブプロセスにてメモリ37からコプロセッサ38にデータを送り終わったことを示す終了コードを受けた場合、ステップST114において終了フラグを生成し、ステップST102の処理に戻る。一方、連続命令コード発行の終了フラグが生成されていない場合には、ステップST109の処理に戻る。

【0068】また、ステップST115において、リードした命令がプロセッサ31の終了命令でないと判断した場合には、プロセッサ31は、ステップST116においてリードした命令を実行する。そして、ステップST117において、プロセッサ31の命令のカウンタであるプログラムカウンタをインクリメントし、ステップST102の処理に戻る。また、ステップST118において、コプロセッサ38の動作が終了しているのを確認して、画像データを出力する。

【0069】以上の処理により、プロセッサ31はコプロセッサ38への命令を連続に発行することで、図6に示すように、並列に動作することが可能になる。

【0070】以上説明したように、本実施形態によれば、プロセッサ31、およびコプロセッサ38で 사용되는命令、データ、並びにコプロセッサ31で直接読み込む命令、データをメモリ37に保持しておき、プロセッサ31において、与えられた命令が、コプロセッサ38で実行される命令である場合には当該命令に対する所定の処理を行って出力し、与えられた命令がコプロセッサ38で直接読み込む命令の場合には、直接読み込み用フラグを設定し、直接読み込み用フラグが設定されていない場合にはプロセッサ31から出力された命令をコプロセッサ38に供給し、直接読み込み用フラグが設定されている場合には、メモリ37の直接読み込み命令、データをコプロセッサ38に直接供給するようにしたので、コプロセッサの命令発行中にプロセッサで別の演算が可能となるため、同一処理時間内にプロセッサで多くの演算を実現できる。プロセッサを通さずに命令を発行できるため、コプロセッサに早く命令を発行できる。コプロセッサに対して命令の発行数が増えるため、多くの画像処理演算を実現できる。その結果、画像処理時間を短縮できる。

【0071】

【発明の効果】以上説明したように、本発明によれば、コプロセッサの命令コード発行中にプロセッサで別の演

算が可能となる。このため、同一処理時間内にプロセッサで多くの演算を実現できる。また、プロセッサを通さずに命令を発行できるため、コプロセッサに早く命令コードを発行できる。その結果、コプロセッサに対して命令コードの発行数が増えるため、多くの画像処理演算を実現でき、ひいては画像処理時間を短縮できる利点がある。

【図面の簡単な説明】

【図1】本発明に係る画像処理装置の一実施形態を示すシステム構成図である。

【図2】本発明に係るメモリのデータの保持の一例を示す図である。

【図3】本発明に係るコプロセッサの構成例を示すブロック図である。

【図4】本発明に係る画像処理装置の全体の動作の流れであるメインプロセスを示すフローチャートである。

【図5】メインプロセスが実行中に並列して動作するメモリからコプロセッサへのデータの供給の動作を表すサブプロセスを示すフローチャートである。

【図6】本発明に係る画像処理装置の命令の実行順序の一例を示す図である。

【図7】SIMD型画像処理プロセッサの構成例を示すブロック図である。

【図8】プロセッサ、およびコプロセッサを用いた従来の画像処理装置のシステム構成例を示すブロック図である。

【図9】図8の画像処理装置におけるメインプロセスについて説明するためのフローチャートである。

【図10】従来の画像処理装置の命令の実行順序の一例を示す図である。

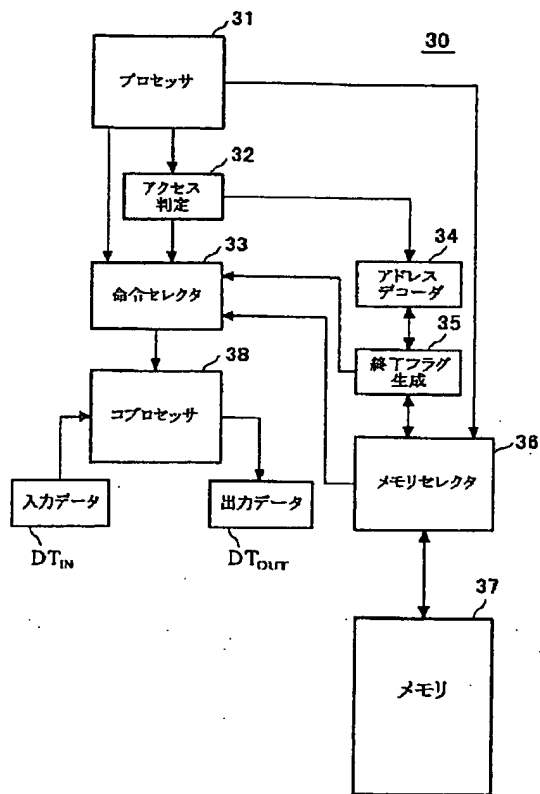
【符号の説明】

30…画像処理装置、31…プロセッサ、32…アクセス判定回路、33…命令セクタ、34…アドレスデコード、35…終了フラグ生成回路、36…メモリセクタ、37…メモリ、38…コプロセッサ、380…SIMD型画像処理プロセッサ、381…入力ポインタ、382a…第1入力SAM部、382b…第2入力SAM部、383…データメモリ部、384…ALUアレイ部、385a…第1出力SAM部、385b…第2出力SAM部、386…出力ポインタ、387…プログラム制御部。

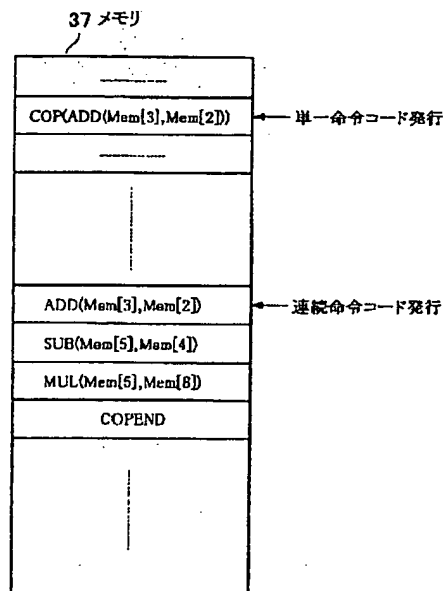
【図10】

命令1	命令2	命令3	コプロ 命令1	コプロ 命令2	命令4	命令5	コプロ 命令3	コプロ 命令4	コプロ 命令5	命令6	命令7
-----	-----	-----	------------	------------	-----	-----	------------	------------	------------	-----	-----

【図1】

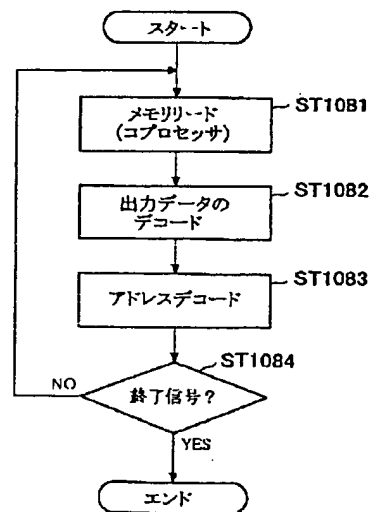


【図2】

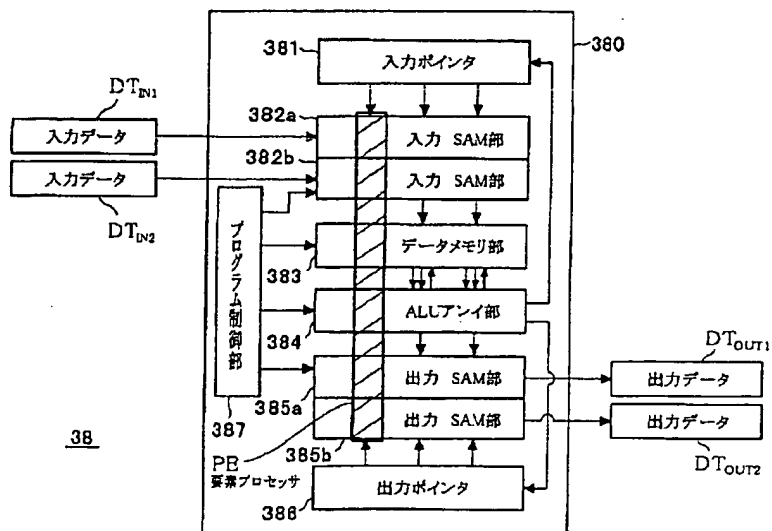


【図5】

## サブプロセス

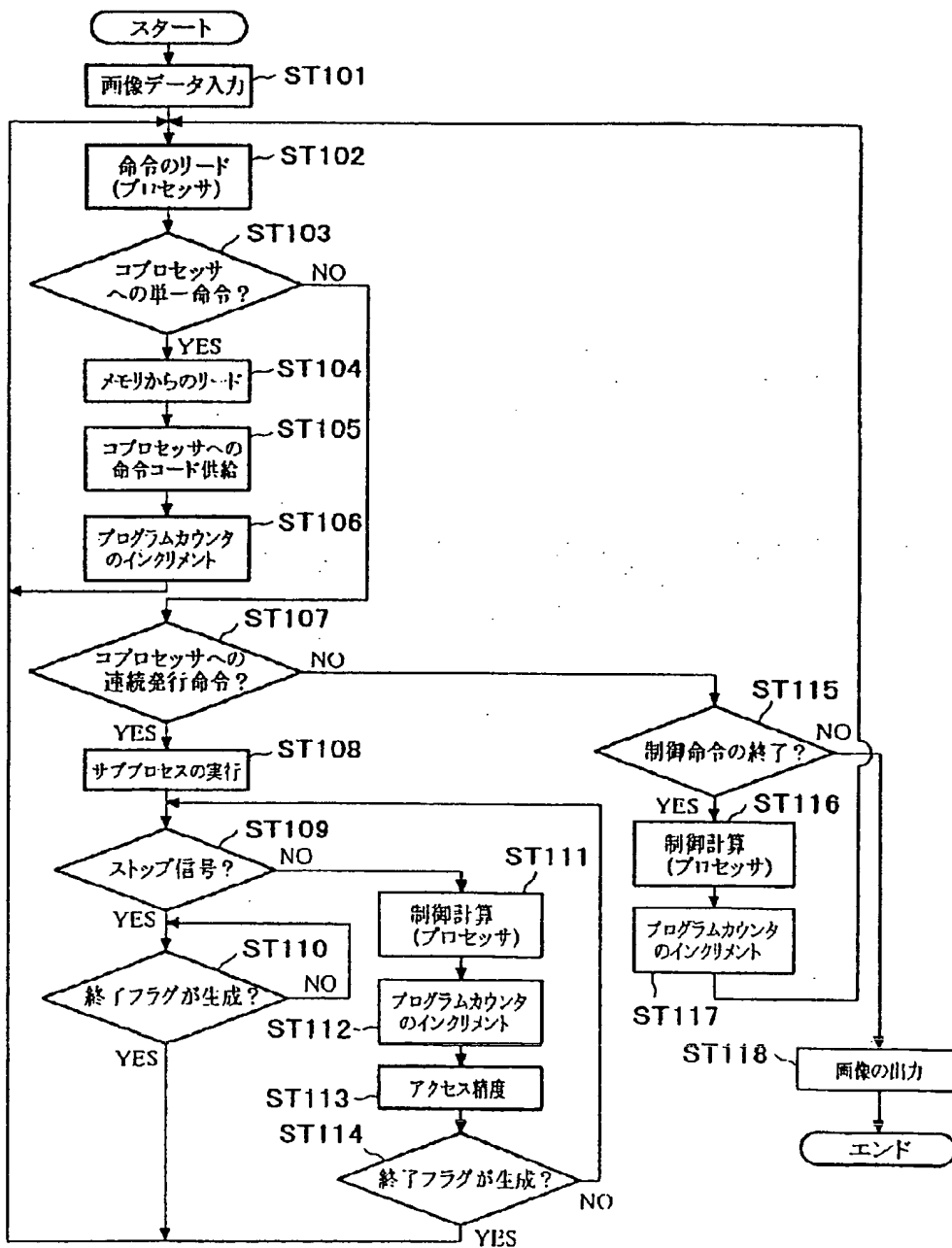


【図3】

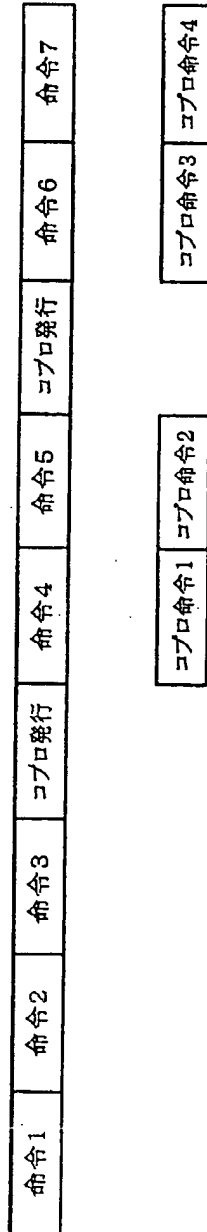


【図4】

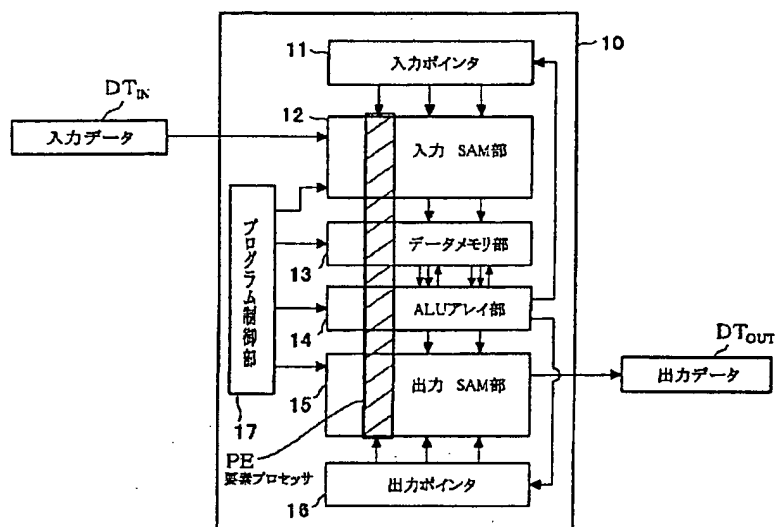
## メインプロセス



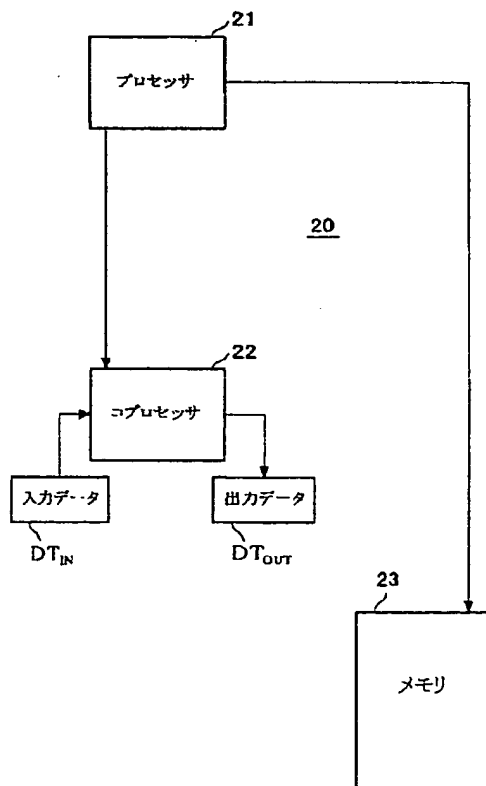
【図6】



【図7】



【図8】



【図9】

